

Matteo Baccan

Costruiamo la nostra Stazione Meteo

Setup di un Raspberry Pi

E' appena arrivato il Raspberry Pi che ho ordinato. Quali sono le prime operazioni da fare per installarlo e configurarlo?

matteo@baccan.it – <http://www.baccan.it>



Matteo Baccan

Divulgatore informatico, giornalista per riviste tecniche nazionali oltre 700 articoli per: Dev, Login, Computer Programming, Mokabyte. Relatore tecnico ad eventi di programmazione: Borland Forum 2000, Webbit 2004, JIP day 2005, Javaday (2006,2007,2010) PHPDay (2008, 2010), CONFSL 2010, WebTech 2010, Codemotin (2011,2012,2013,2014). Autore di Corso di C# ISBN 8881500167. Autore di JobCrawler e HTML2POP3 (Oltre 900.000 download su SourceForge)

Il mio motto

**Per fare un grande piatto devi togliere
non aggiungere**

*Gualtiero Marchesi
(a volte vale anche per un grande software)*

Cosa vedremo

- Prima configurazione di Raspberry Pi
- Ottimizzazioni in pratica
- Installazione ambiente LAMP
- Installazione sensori
- Lettura dei dati dai sensori
- Creazione ambiente virtuale

Quali passi seguire dopo aver creato l'SD con Raspbian?

- Personalizzazione lingua
- Ottimizzazioni Raspbian in base all'uso di Raspberry Pi
- Installazione aggiornamenti e pulizia pacchetti
- Installazione ambiente LAMP

Primo accesso

```
not closed circuit  
[ ok ] Starting OpenBSD Secure Shell server: sshd.  
My IP address is 10.0.2.15  
  
Raspbian GNU/Linux 7 raspberrypi tty1  
  
raspberrypi login:
```

Alla prima esecuzione, l'utente d'accesso su Raspberry è

utente: pi

password: raspberry

Raspbian nasce in lingua inglese, per poterlo usare in italiano occorre fare una serie di modifiche con

```
sudo raspi-config
```

Raspberry Pi Software Configuration Tool (raspi-config)

- | | |
|---------------------------------------|--|
| 1 Expand Filesystem | Ensures that all of the SD card s |
| 2 Change User Password | Change password for the default u |
| 3 Enable Boot to Desktop/Scratch | Choose whether to boot into a des |
| 4 Internationalisation Options | Set up language and regional sett |
| 5 Enable Camera | Enable this Pi to work with the R |
| 6 Add to Rastrack | Add this Pi to the online Raspber |
| 7 Overclock | Configure overclocking for your P |
| 8 Advanced Options | Configure advanced settings |
| 9 About raspi-config | Information about this configurat |

<Select>

<Finish>

Cambiamo quindi lingua e impostazioni

```
Raspberry Pi Software Configuration Tool (raspi-config)

I1 Change Locale          Set up language and regional sett
I2 Change Timezone        Set up timezone to match your loc
I3 Change Keyboard Layout Set the keyboard layout to match

<Select>                  <Back>
```


Configuring locales

Locales are a framework to switch between multiple languages and allow users to use their language, country, characters, collation order, etc.

Please choose which locales to generate. UTF-8 locales should be chosen by default, particularly for new installations. Other character sets may be useful for backwards compatibility with older systems and software.

Locales to be generated:

```
[ ] is_IS.UTF-8 UTF-8
[ ] it_CH.ISO-8859-1
[ ] it_CH.UTF-8 UTF-8
[ ] it_IT.ISO-8859-1
[*] it_IT.UTF-8 UTF-8
[ ] it_IT@euro ISO-8859-15
[ ] iu_CA.UTF-8
[ ] iw_IL.ISO-8859-8
[ ] iw_IL.UTF-8 UTF-8
[ ] ja_JP.EUC-JP EUC-JP
[ ] ja_JP.UTF-8 UTF-8
```

<Ok>

<Cancel>

Timezone

Configuring tzdata

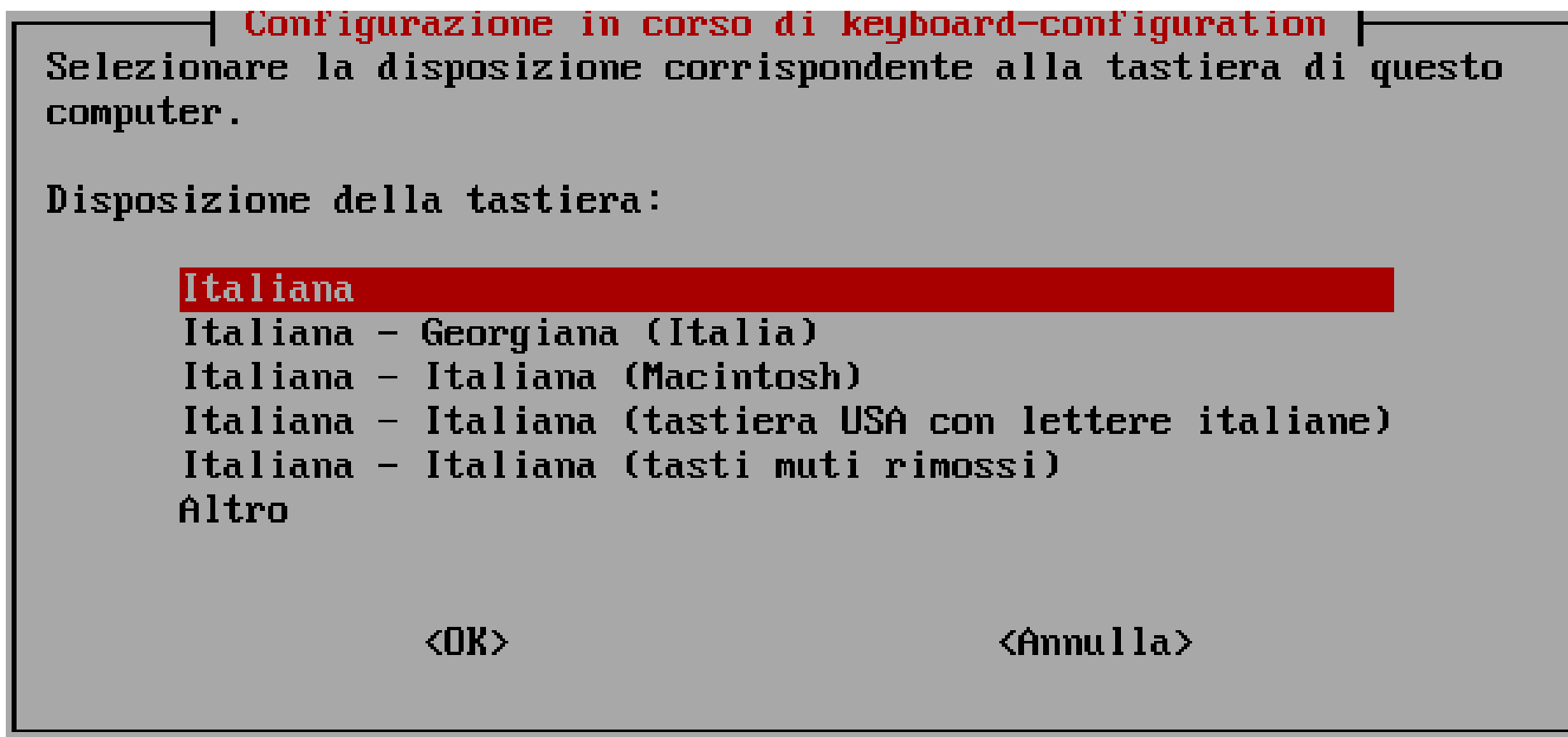
Please select the city or region corresponding to your time zone.

Time zone:

London	↑
Luxembourg	
Madrid	
Malta	
Mariehamn	
Minsk	
Monaco	
Moscow	
Nicosia	
Oslo	
Paris	
Podgorica	
Prague	
Riga	
Rome	
Samara	↓

<Ok> <Cancel>

e se non vogliamo avere problemi con le tastiere italiane

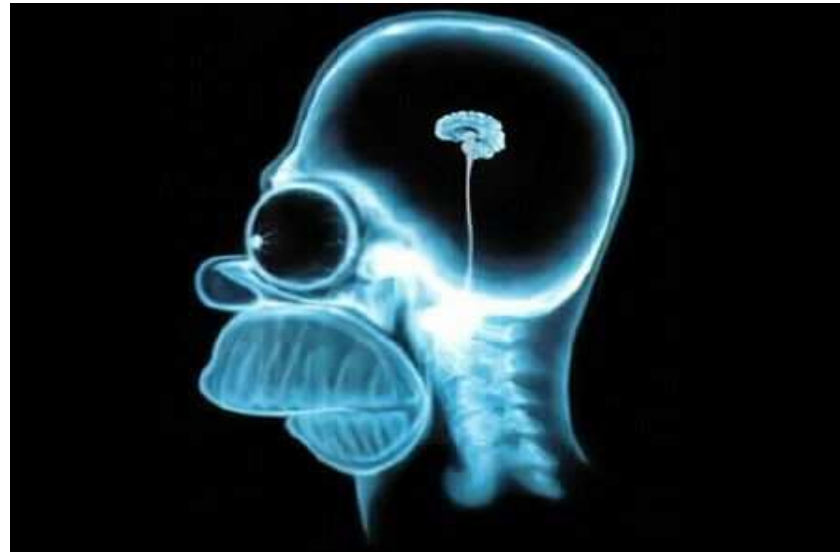


Passiamo quindi a 2 ottimizzazioni veloci per aumentare i due aspetti maggiormente toccati da applicazioni LAMP

Velocità



Memoria



Overclock a 1000MHz

Chose overclock preset

None	700MHz ARM, 250MHz core, 400MHz SDRAM, 0 overvolt
Modest	800MHz ARM, 250MHz core, 400MHz SDRAM, 0 overvolt
Medium	900MHz ARM, 250MHz core, 450MHz SDRAM, 2 overvolt
High	950MHz ARM, 250MHz core, 450MHz SDRAM, 6 overvolt
Turbo	1000MHz ARM, 500MHz core, 600MHz SDRAM, 6 overvolt
Pi2	1000MHz ARM, 500MHz core, 500MHz SDRAM, 2 overvolt

<OK>

<Annulla>

Memory Split a 16MB

How much memory should the GPU have? e.g. 16/32/64/128/256

16

<OK> <Annulla>

Eventuali approfondimenti sui parametri modificabili tramite raspi-config sono recuperabili a questo indirizzo

<http://www.raspberrypi.org/documentation/configuration/raspi-config.md>

`raspi-config` is the Raspberry Pi configuration tool written and maintained by [Alex Bradbury](#). It targets Raspbian.

La configurazione di Raspbian avviene tramite APT: Advanced Packaging Tool

Aggiornare l'indice dei pacchetti
`sudo apt-get update`

Aggiornare i pacchetti
`sudo apt-get upgrade`



A fine installazione ci sono dei passi che possono aiutarci
ad avere più spazio nel sistema

Rimuovere wolfram-engine che occupa parecchio

```
sudo apt-get remove wolfram-engine
```

Pulizia di pacchetti non più utilizzati

```
sudo apt-get autoremove
```

Recupero lo spazio usato dai pacchetti scaricati e installati

```
sudo apt-get clean
```

A questo punto abbiamo un Raspberry Pi aggiornato e pronto per poter essere utilizzato come un server LAMP



Installazione stack LAMP

Di default, lo **stack LAMP non è installato** su Raspbian.

Raspbian parte infatti come sistema operativo “client” e non “server”, per trasformarlo in server occorre installare quello che manca, o scaricare una distribuzione apposita

Essendo un sistema Linux, l’installazione dello stack LAMP è identica a quella che si farebbe su un qualsiasi server Debian non Raspberry Pi

Installiamo lo stack LAMP + PHPMyAdmin

Tramite **apt**, possiamo installare

- Apache + l'integrazione a php
- PHP5 + l'interfaccia verso mysql + la versione fpm di php
- MySQL server e client
- PhpMyAdmin

```
sudo apt-get install apache2 php5 mysql-server  
libapache2-mod-php5 php5-mysql php5-fpm mysql-client  
phpmyadmin
```

Installiamo lo stack LAMP + PHPMyAdmin

Configurazione in corso di mysql-server-5.5

Sebbene non sia obbligatoria, si raccomanda d'impostare una password per l'utente d'amministrazione «root» di MySQL.

Se questo campo è lasciato vuoto, la password non viene cambiata.

Nuova password per l'utente «root» di MySQL:

<OK>

Installiamo lo stack LAMP + PHPMyAdmin

Configurazione in corso di phpmyadmin

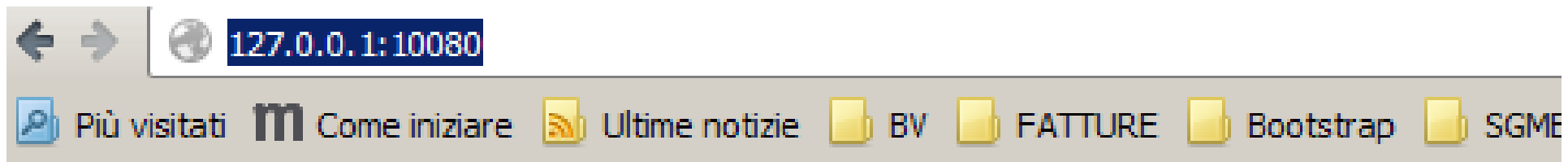
Scegliere il server web da configurare automaticamente per eseguire phpMyAdmin.

Server web da riconfigurare automaticamente:

☒ apache2
☐ lighttpd

<OK>

Proviamo Apache



It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

Proviamo PHPMyAdmin



← → PMA 127.0.0.1:10080/phpmyadmin/

phpMyAdmin

Benvenuto in phpMyAdmin

Lingua - Language

Italiano - Italian ▼

Connetti 🔑

Nome utente:

Password:

A fine installazione, dopo le pulizie di pacchetti e l'installazione di LAMP, Raspbian occupa circa 2.2GB

```
pi@raspberrypi ~ $ df -h
File system      Dim. Usati Dispon. Uso% Montato su
rootfs           2,9G  2,2G   623M   78% /
/dev/root        2,9G  2,2G   623M   78% /
devtmpfs         125M    0   125M    0% /dev
tmpfs            25M   216K    25M    1% /run
tmpfs            5,0M    0    5,0M    0% /run/lock
tmpfs            50M    0    50M    0% /run/shm
/dev/sda1        56M   15M    42M   26% /boot
```

Per la realizzazione della stazione meteo useremo il seguente sensore di: pressione, altitudine e temperatura

I2C BMP085 Digital Barometric Pressure Sensor Module Barometer sensor 3-5V

US \$7,84

Circa EUR 6,88

Compralo Subito

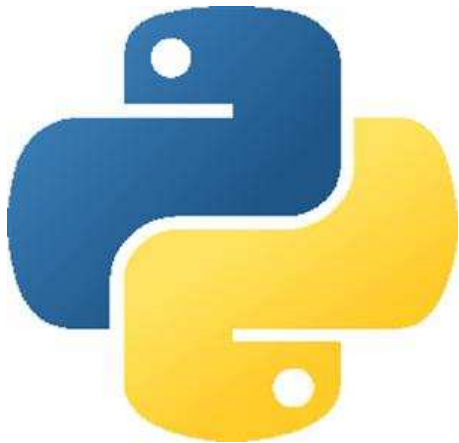
[Aggiungi al carrello](#)



I sensori usati in questa esercitazione sono forniti da Adafruit Industries.

Oltre a fornire i sensori, Adafruit fornisce anche alcuni esempi Python di come leggere dei dati rilevati dai sensori stessi.

Per questo motivo, abbiamo optato per una personalizzazione di tali programmi, in modo che non si limitassero a leggere i dati e a visualizzarli, ma che li andassero anche a scrivere su un database MySQL.



Adafruit fornisce il codice di lettura dei sensori in OpenSource

È disponibile un repository su GitHub, con una serie di programmi già pronti per la lettura di una serie di componenti

`https://github.com/adafruit/Adafruit-Raspberry-Pi-Python-Code`

Nel nostro caso Adafruit_BMP085

Componenti gestiti

Adafruit_ADS1x15
Adafruit_ADXL345
Adafruit_BMP085
Adafruit_CharLCD
Adafruit_CharLCDPlate
Adafruit_DHT_Driver
Adafruit_DHT_Driver_Python
Adafruit_I2C
Adafruit_LEDBackpack
Adafruit_LEDpixels
Adafruit_LSM303
Adafruit_MCP230xx
Adafruit_MCP3002
Adafruit_MCP3008
Adafruit_MCP4725
Adafruit_PWM_Servo_Driver
Adafruit_TCS34725
Adafruit_VCNL4000

Script di base di Adafruit

```
#!/usr/bin/python

from Adafruit_BMP085 import BMP085

# Initialise the BMP085 and use STANDARD mode (default value)
# bmp = BMP085(0x77, debug=True)
bmp = BMP085(0x77)

temp = bmp.readTemperature()

# Read the current barometric pressure level
pressure = bmp.readPressure()

# To calculate altitude based on an estimated mean sea level pressure
# (1013.25 hPa) call the function as follows, but this won't be very accurate
altitude = bmp.readAltitude()

# To specify a more accurate altitude, enter the correct mean sea level
# pressure level. For example, if the current pressure level is 1023.50 hPa
# enter 102350 since we include two decimal places in the integer value
# altitude = bmp.readAltitude(102350)

print "Temperature: %.2f C" % temp
print "Pressure:      %.2f hPa" % (pressure / 100.0)
print "Altitude:      %.2f" % altitude
```

Script modificato

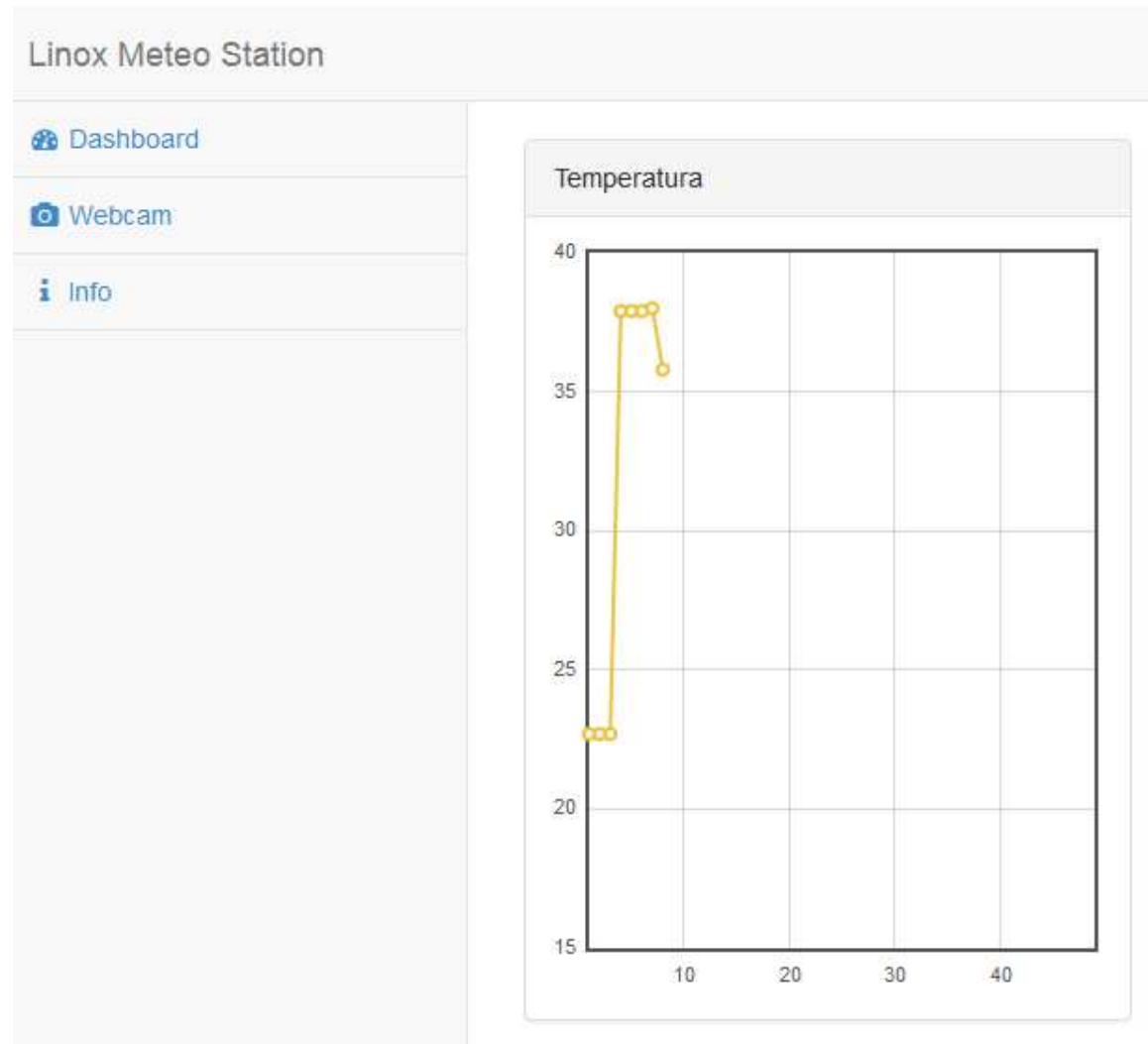
```
bmp = BMP085(0x77)

temp = bmp.readTemperature()
pressure = bmp.readPressure()
altitude = bmp.readAltitude()

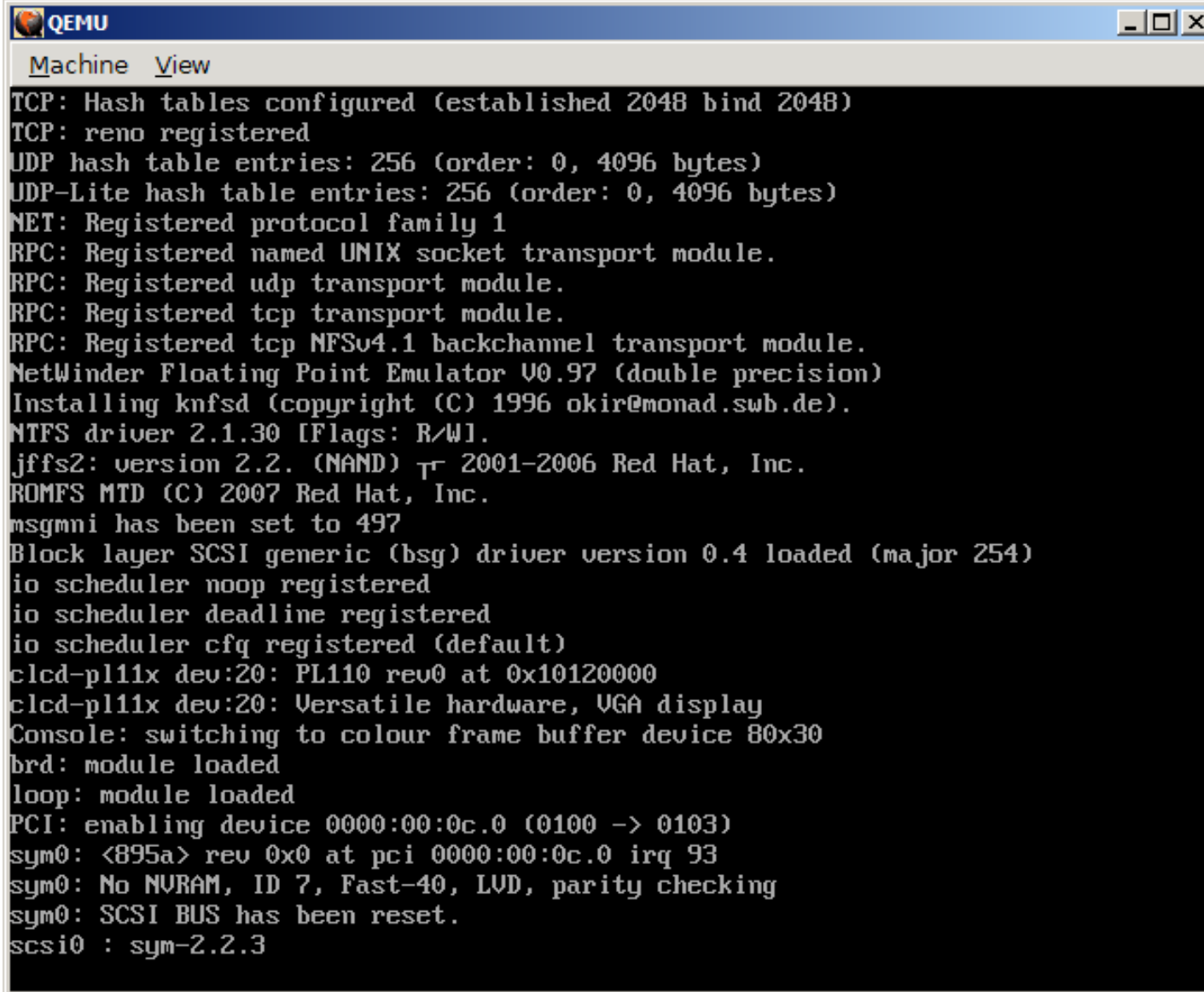
insert = "insert into
sensori(stazione,data,temperatura,pressione,altitudine)
values('Linux','%s',%.2f,%.2f,%.2f);" %
(datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S"), temp ,
(pressure / 100.0) , altitude)

con = mdb.connect('127.0.0.1', 'pi', 'raspberrypi', 'meteo')
cur = con.cursor()
cur.execute( insert )
```

Demo



Quali passi seguire per ottenere un Raspberry Pi virtuale ?

A screenshot of a QEMU window titled "QEMU". The window has a menu bar with "Machine" and "View". The main area displays a series of kernel boot logs in a monospaced font. The logs show the initialization of various components including TCP/UDP hash tables, RPC modules, NetWinder floating point emulator, knfsd, NTFS driver, jffs2, ROMFS MTD, msgmni, block layer SCSI generic driver, io schedulers, clcd-pl11x display, console switching, and PCI/Symtec SCSI controller.

```
QEMU
Machine View
TCP: Hash tables configured (established 2048 bind 2048)
TCP: reno registered
UDP hash table entries: 256 (order: 0, 4096 bytes)
UDP-Lite hash table entries: 256 (order: 0, 4096 bytes)
NET: Registered protocol family 1
RPC: Registered named UNIX socket transport module.
RPC: Registered udp transport module.
RPC: Registered tcp transport module.
RPC: Registered tcp NFSv4.1 backchannel transport module.
NetWinder Floating Point Emulator V0.97 (double precision)
Installing knfsd (copyright (C) 1996 okir@monad.swb.de).
NTFS driver 2.1.30 [Flags: R/W].
jffs2: version 2.2. (NAND) © 2001-2006 Red Hat, Inc.
ROMFS MTD (C) 2007 Red Hat, Inc.
msgmni has been set to 497
Block layer SCSI generic (bsg) driver version 0.4 loaded (major 254)
io scheduler noop registered
io scheduler deadline registered
io scheduler cfq registered (default)
clcd-pl11x dev:20: PL110 rev0 at 0x10120000
clcd-pl11x dev:20: Versatile hardware, VGA display
Console: switching to colour frame buffer device 80x30
brd: module loaded
loop: module loaded
PCI: enabling device 0000:00:0c.0 (0100 -> 0103)
sym0: <895a> rev 0x0 at pci 0000:00:0c.0 irq 93
sym0: No NVRAM, ID 7, Fast-40, LVD, parity checking
sym0: SCSI BUS has been reset.
scsi0 : sym-2.2.3
```

Sul sito xecdesign.com è disponibile una guida passo passo per realizzare una versione virtuale di Raspbian

<http://xecdesign.com/qemu-emulating-raspberry-pi-the-easy-way/>

Il procedimento è molto semplice e può ridursi a pochi semplici passi da seguire.

Il primo passo è quello di recuperare i 3 componenti software:

- L'immagine del kernel Linux scaricabile da [xecdesign](http://xecdesign.com)

L'immagine di RASPBIAN

<http://www.raspberrypi.org/downloads>



RASPBIAN
Debian Wheezy

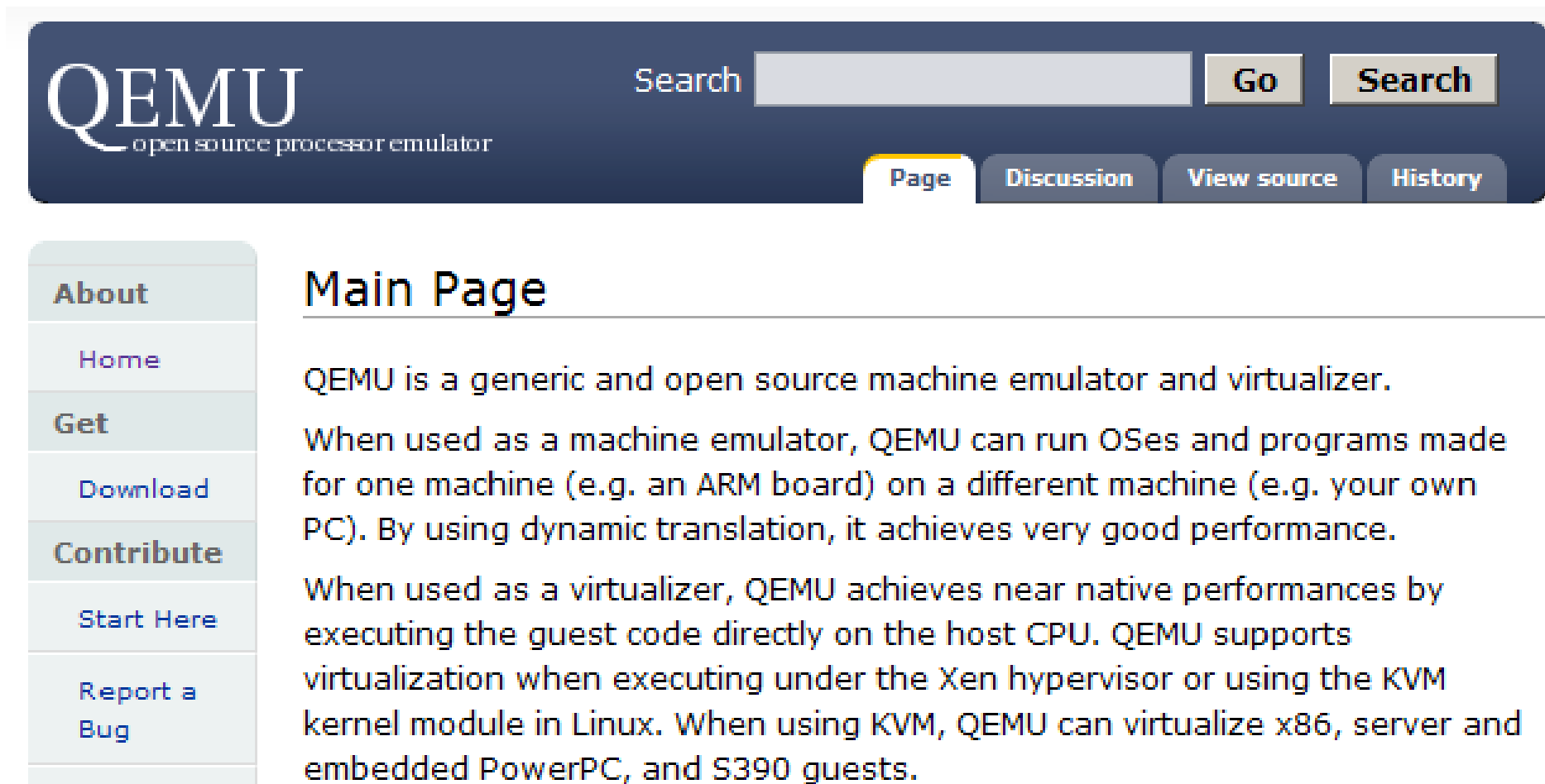
Version: January 2015
Release date: 2015-01-31

[More info +](#)

[Download Torrent](#) [Download ZIP](#)

L'emulatore QEMU

<http://www.qemu.org>



The screenshot shows the QEMU website's main page. At the top, there's a dark blue header with the QEMU logo on the left, which includes the text "QEMU" in a large serif font and "open source processor emulator" in a smaller sans-serif font below it. To the right of the logo is a search bar with the placeholder text "Search", a "Go" button, and a "Search" button. Below the search bar are four tabs: "Page" (which is highlighted with a yellow border), "Discussion", "View source", and "History". On the left side of the page, there's a vertical sidebar with a light blue background. It contains several links: "About", "Home" (in purple), "Get", "Download", "Contribute", "Start Here", "Report a Bug", and a blank link at the bottom. The main content area on the right has the title "Main Page" in a large, bold, black serif font, followed by a horizontal line. Below the title, there are two paragraphs of text in a black serif font. The first paragraph states that QEMU is a generic and open source machine emulator and virtualizer. The second paragraph describes its use as a machine emulator, noting that it can run OSes and programs made for one machine (e.g., an ARM board) on a different machine (e.g., your own PC) by using dynamic translation to achieve very good performance. The third paragraph describes its use as a virtualizer, noting that it achieves near native performances by executing the guest code directly on the host CPU. It also mentions that QEMU supports virtualization when executing under the Xen hypervisor or using the KVM kernel module in Linux, and that it can virtualize x86, server and embedded PowerPC, and S390 guests.

QEMU
open source processor emulator

Search **Go** **Search**

Page Discussion View source History

About

Home

Get

Download

Contribute

Start Here

Report a Bug

Main Page

QEMU is a generic and open source machine emulator and virtualizer.

When used as a machine emulator, QEMU can run OSes and programs made for one machine (e.g. an ARM board) on a different machine (e.g. your own PC). By using dynamic translation, it achieves very good performance.

When used as a virtualizer, QEMU achieves near native performances by executing the guest code directly on the host CPU. QEMU supports virtualization when executing under the Xen hypervisor or using the KVM kernel module in Linux. When using KVM, QEMU can virtualize x86, server and embedded PowerPC, and S390 guests.

Una volta recuperati emulatore, immagine e kernel, possiamo procedere al primo boot

```
qemu-system-armw -kernel kernel-qemu-3.10.26+ -cpu arm1176 -m 256 -M versatilepb  
-no-reboot -serial stdio -append "root=/dev/sda2 panic=1 rootfstype=ext4 rw  
init=/bin/bash" -hda 2015-01-31-raspbian.img
```

Dal quale potremo andare a modificare un paio di file di configurazione.

Prima commentiamo una riga in ld.so.preload

```
nano /etc/ld.so.preload  
#/usr/lib/arm-linux-gnueabi/libcofi_rpi.so
```

Poi creiamo il file 90-qemu.rules

```
nano /etc/udev/rules.d/90-qemu.rules  
KERNEL=="sda", SYMLINK+="mmcblk0"  
KERNEL=="sda?", SYMLINK+="mmcblkop%n"  
KERNEL=="sda2", SYMLINK+="root"
```

Infine terminiamo la sessione

```
sudo halt
```

Alla seconda esecuzione, possiamo procedere con un boot normale, rimuovendo il parametro

init=/bin/bash

Che ci permettesse un login con bash, necessario alla modifica dei soli file di configurazione

Vi consiglio anche qualche parametro per poter gestire la macchina virtuale dalla macchina host, reindirizzando le porte **22 (ssh)** e **80 (http)**

```
set QEMU_NET=-net nic -net user,hostfwd=tcp::10022-:22,hostfwd=tcp::10080-:80
```

```
qemu-system-armw %QEMU_NET% -kernel kernel-qemu-3.10.26+ -cpu arm1176 -m  
256 -M versatilepb -no-reboot -serial stdio -append "root=/dev/sda2 panic=1  
rootfstype=ext4 rw" -hda 2015-01-31-raspbian.img
```

DEMO

Per chi volesse approfondire

Raspberry Pi foundation: <http://www.raspberrypi.org>

Raspbian: <http://www.raspberrypi.org/downloads>

Media center Raspmbc: <http://www.raspbmc.com>

Chameleon: <http://chameleon.enging.com>

A Pen Test Drop Box Distro: <http://www.pwnpi.com/>

Guida all'emulazione di Raspberry: <http://xecdesign.com>

Linux Novara: <http://www.linuxnovara.org/>





Matteo Baccan

matteo@baccan.it

<http://www.baccan.it>